

Amendments to the Claims

1 · **Claim 1 (currently amended): A computer-implemented method of programmatically generating**  
2 **a class library to represent messages described in a structured language specification, comprising**  
3 **steps of:**

4 **detecting, during run-time processing of a machine-processable definition of a network-**  
5 **invocable service, a reference to a structured language specification;**

6 **locating, responsive to the detection, the referenced structured language specification, the**  
7 **structured language specification encoded in a structured markup language and specifying**  
8 **message syntax definitions for one or more messages usable for interacting with the network-**  
9 **invocable service; and**

10 ~~locating, responsive to the detection, a language-specific template that specifies an image~~  
11 ~~for generating code as a class library for a particular coding language and specifies where~~  
12 ~~corresponding portions of message syntax definitions are to be substituted therein; and~~  
13 ~~generating [[the]] code for the message syntax definitions in the located structured~~  
14 ~~language specification, according to code-generation guidance specified in a dynamically-~~  
15 ~~selected one of a plurality of language-specific code-generation templates that each specify~~  
16 ~~guidance for generating code in a different programming language, the guidance specified as an~~  
17 ~~image of code to be generated in that programming language and comprising syntax indicating~~  
18 ~~where portions of the message syntax definitions are to be substituted for portions of the~~  
19 ~~specified image, wherein the generated code for the message syntax definitions comprises the~~  
20 ~~template and the definitions in the structured language specification, comprising the class library,~~  
21 ~~such that instances of classes specified by the class library are instantiable to be dynamically~~

22 available for sending request messages to, and receiving response messages from, the network-  
23 invocable service, further comprising steps of:

24 locating, in the structured language specification, the message syntax definitions  
25 of the messages;

26 applying the selected template to the located message syntax definitions to  
27 generate code that, when executed, will build an instance of [[the]] each message for sending and  
28 will, if the message syntax definition for the message specifies parameters, dynamically obtain  
29 values for the parameters and set those parameter values in the built instance;

30 applying the selected template to the located message syntax definitions to  
31 generate code that, when executed, will send the built instance of [[the]] each message, including  
32 any set parameter values, to the network-invocable service as a request message;

33 applying the selected template to the located message syntax definitions to  
34 generate code that, when executed, will receive a response to the sent instance of [[the]] each  
35 message from the network-invocable service as a response message and build a response instance  
36 therefrom; and

37 applying the selected template to the located message syntax definitions to  
38 generate code that, when executed, will dynamically obtain any defined response values from  
39 [[the]] each received response message and populate the response instance therewith;

40 such that the dynamically-generated programmatically-generated code is dynamically  
41 invocable during the run-time processing for sending the request messages to, and receiving the  
42 response messages from, the network-invocable service.

1       **Claim 2 (previously presented): The method according to Claim 1, wherein the structured**  
2       **language specification is a schema.**

1       **Claim 3 (previously presented): The method according to Claim 1, wherein the structured**  
2       **language specification is a Document Type Definition ("DTD").**

1       **Claim 4 (original): The method according to Claim 1, wherein the structured markup language is**  
2       **Extensible Markup Language ("XML").**

1       **Claim 5 (previously presented): The method according to Claim 1, wherein the message syntax**  
2       **definitions specify elements corresponding to the messages and optionally specify attributes**  
3       **corresponding to the elements, the elements and attributes being encoded in the structured**  
4       **markup language.**

1       **Claim 6 (previously presented): The method according to Claim 5, wherein the message syntax**  
2       **definitions specify, for at least one of the elements, one or more child elements.**

1       **Claim 7 (previously presented): The method according to Claim 5, wherein the message syntax**  
2       **definitions specify whether the attributes are required attributes.**

Claims 8 - 15 (canceled)

1       Claim 16 (previously presented): The method according to Claim 1, further comprising the step  
2       of programmatically consulting one or more rules, wherein the rules specify one or more of (1)  
3       where the generated code should be stored and (2) a name for a class library comprising the  
4       generated code, to influence processing of the generating step.

Claims 17 - 19 (canceled)

1       Claim 20 (previously presented): The method according to Claim 1, wherein the network-  
2       invocable service is a web service.

1       Claim 21 (previously presented): The method according to Claim 20, wherein the reference is  
2       specified as a Uniform Resource Locator and the machine-processable definition is specified in a  
3       Web Services Definition Language document.

Claim 22 - 25 (canceled)

1       Claim 26 (currently amended): A system for programmatically generating a class library to  
2       represent messages described in a structured language specification, comprising:  
3               means for detecting, during run-time processing of a machine-processable definition of a  
4       network-invocable service, a reference to a structured language specification;  
5               means for locating, responsive to the detection, the referenced structured language  
6       specification, the structured language specification encoded in a structured markup language and

7 specifying message syntax definitions for one or more messages usable for interacting with the  
8 network-invocable service; and  
9 —— means for locating, responsive to the detection, a language-specific template that specifies  
10 an image for generating code as a class library for a particular coding language and specifies  
11 where corresponding portions of message syntax definitions are to be substituted therein; and  
12 means for generating [[the]] code for the message syntax definitions in the located  
13 structured language specification, according to code-generation guidance specified in a  
14 dynamically-selected one of a plurality of language-specific code-generation templates that each  
15 specify guidance for generating code in a different programming language, the guidance specified  
16 as an image of code to be generated in that programming language and comprising syntax  
17 indicating where portions of the message syntax definitions are to be substituted for portions of  
18 the specified image, wherein the generated code for the message syntax definitions comprises the  
19 template and the definitions in the structured language specification, comprising the class library,  
20 such that instances of classes specified by the class library are instantiable to be dynamically  
21 available for sending request messages to, and receiving response messages from, the network-  
22 invocable service, further comprising:  
23       means for locating, in the structured language specification, the message syntax  
24       definitions of the messages;  
25       means for applying the selected template to the located message syntax definitions  
26       to generate code that, when executed, will build an instance of [[the]] each message for sending  
27       and will, if the message syntax definition for the message specifies parameters, dynamically  
28       obtain values for the parameters and set those parameter values in the built instance;

29 means for applying the selected template to the located message syntax definitions  
30 to generate code that, when executed, will send the built instance of [[the]] each message,  
31 including any set parameter values, to the network-invocable service as a request message;  
32 means for applying the selected template to the located message syntax definitions  
33 to generate code that, when executed, will receive a response to the sent instance of [[the]] each  
34 message from the network-invocable service as a response message and build a response instance  
35 therefrom; and  
36 means for applying the selected template to the located message syntax definitions  
37 to generate code that, when executed, will dynamically obtain any defined response values from  
38 [[the]] each received response message and populate the response instance therewith;  
39 such that the dynamically-generated programmatically-generated code is dynamically  
40 invocable during the run-time processing for sending the request messages to, and receiving the  
41 response messages from, the network-invocable service.

1 Claim 27 (currently amended): A computer program product for programmatically generating a  
2 class library to represent messages described in a structured language specification, the computer  
3 program product embodied on one or more computer-usable media and comprising:  
4 computer-readable program code [[means]] for detecting, during run-time processing of a  
5 machine-processable definition of a network-invocable service, a reference to a structured  
6 language specification;  
7 computer-readable program code [[means]] for locating, responsive to the detection, the  
8 referenced structured language specification, the structured language specification encoded in a

9       structured markup language and specifying message syntax definitions for one or more messages  
10      usable for interacting with the network-invocable service; and  
11      — computer-readable program code means for locating, responsive to the detection, a  
12      language-specific template that specifies an image for generating code as a class library for a  
13      particular coding language and specifies where corresponding portions of message syntax  
14      definitions are to be substituted therein; and  
15            computer-readable program code [[means]] for generating [[the]] code for the message  
16      syntax definitions in the located structured language specification, according to code-generation  
17      guidance specified in a dynamically-selected one of a plurality of language-specific code-  
18      generation templates that each specify guidance for generating code in a different programming  
19      language, the guidance specified as an image of code to be generated in that programming  
20      language and comprising syntax indicating where portions of the message syntax definitions are  
21      to be substituted for portions of the specified image, wherein the generated code for the message  
22      syntax definitions comprises the template and the definitions in the structured language  
23      specification, comprising the class library, such that instances of classes specified by the class  
24      library are instantiable to be dynamically available for sending request messages to, and receiving  
25      response messages from, the network-invocable service, further comprising:

26            computer-readable program code [[means]] for locating, in the structured  
27      language specification, the message syntax definitions of the messages;  
28            computer-readable program code [[means]] for applying the selected template to  
29      the located message syntax definitions to generate code that, when executed, will build an  
30      instance of [[the]] each message for sending and will, if the message syntax definition for the

31 message specifies parameters, dynamically obtain values for the parameters and set those  
32 parameter values in the built instance;  
33 computer-readable program code [[means]] for applying the selected template to  
34 the located message syntax definitions to generate code that, when executed, will send the built  
35 instance of [[the]] each message, including any set parameter values, to the network-invocable  
36 service as a request message;  
37 computer-readable program code [[means]] for applying the selected template to  
38 the located message syntax definitions to generate code that, when executed, will receive a  
39 response to the sent instance of [[the]] each message from the network-invocable service as a  
40 response message and build a response instance therefrom; and  
41 computer-readable program code [[means]] for applying the selected template to  
42 the located message syntax definitions to generate code that, when executed, will dynamically  
43 obtain any defined response values from [[the]] each received response message and populate the  
44 response instance therewith;  
45 such that the dynamically-generated programmatically-generated code is dynamically  
46 invocable during the run-time processing for sending the request messages to, and receiving the  
47 response messages from, the network-invocable service.

1 Claim 28 (new): The method according to Claim 1, wherein the dynamically-selected one of the  
2 templates is that one of the templates that specifies the guidance in a particular programming  
3 language for which the class library is to be generated.

1       **Claim 29 (new): The method according to Claim 1, wherein each of the templates is independent**  
2       **of the message syntax definitions for which the class library is to be generated.**